REMARKS

Claims 1-11 and 14-17 are all the claims pending in the application. By this Amendment, Applicant amends claims 1, 5, 10, and 14 to further clarify the invention and claims 6 and 16 to cure minor informalities.

I. Summary of the Office Action

The Examiner objected to claims 6, 11, and 16 for minor informalities. In addition, the Examiner rejected claims 1-11 and 14-17 under 35 U.S.C. § 103(a). Specifically,

a) claims 1, 3-5, 11, and 14 are rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent 6,112,024 to Almond et al. (hereinafter "Almond") in view of U.S. Patent 5,453,933 to Wright et al. (hereinafter "Wright") and further in view of the alleged Applicant's Admitted Prior Art, pages 1-10 of the specification (hereinafter "APA");

b) claims 2 and 10 are rejected under 35 U.S.C. § 103(a) as being unpatentable over Almond, Wright, and the APA in view of "Versions and change notification in an object-oriented database system" by Chou et al. (hereinafter "Chou");

c) claims 6, 8, and 9 are rejected under 35 U.S.C. § 103(a) as being unpatentable over Almond, Wright, and the APA in view of U.S. Patent 5,907,705 to Carter (hereinafter "Carter");

d) claim 7 is rejected under 35 U.S.C. 103(a) as being unpatentable over Almond, Wright, the APA, and Carter in view of "Linkers & Loaders" by Levine (hereinafter "Levine"); and

e) claims 15-17 are rejected under 35 U.S.C. § 103(a) as being unpatentable over

Almond, Wright, and the APA in view of a newly found reference, U.S. Patent 5,734,902 to

Atkins et al. (hereinafter "Atkins").

II.   Claim Objections

The Examiner objected to claims 6 and 16 for minor informalities. Applicant respectfully

requests the Examiner to withdraw these objections to claims 6 and 16 in view of the self-

explanatory claim amendments.

The Examiner also objected to claim 11 alleging that it fails to further limit claim 1.

Applicant respectfully traverses this objection in view of the following comments. Claim 1

requires at least two of: a display device, a supervision device, and a programmable controller.

In other words, only two devices are required to meet the unique features of claim 1. Claim 11,

on the other hand, requires three devices (the display device, the supervision device, and the

programmable controller). In other words, claim 11 further limits the subject matter of claim 1.

In view thereof, it is appropriate and necessary for the Examiner to withdraw this objection of

claim 11.

III.   Claim Rejections under 35 USC § 103(a)

The Examiner rejected claims 1-11 and 14-17 under 35 U.S.C. § 103(a). Applicant

respectfully traverses this rejection in view of the following comments.

Almond, Wright, and the APA

Claims 1, 3-5, 11, and 14 are rejected under 35 U.S.C. § 103(a) as being unpatentable

over Almond in view of Wright and the APA.

Applicant respectfully submits that one of ordinary skill in the art would not have been motivated to combine the references in the manner suggested by the Examiner. *Almond* relates to a development system providing methods for managing different versions of objects. The development tools may access an object stored on the object server via RPC. These various software development tools are programming languages such as Powersoft, PowerBuilder, Visual C++, and so on (col. 5, lines 9 to 21).

Wright, on the other hand, relates to a CNC machine tool control system that utilizes an object-oriented software environment where objects are arranged in hierarchy (col. 6, lines 21 to 40). Accordingly, in *Wright*, the CNC machine tool control is easily modifiable. That is, the previously written software can be reused and new objects can be added which inherit or comprise compositions of previously defined objects (col. 5, lines 58 to 62).

The Examiner alleges that the references are analogous because both deal with software development. The Examiner further alleges that there is motivation to combine the references so that "external machines can be controlled in an object oriented control system" (*see* page 4 of the Office Action).

Under the rationale provided in the Office Action, it would appear that *any* computer science reference, no matter how unrelated, is combinable with another computer science reference because they all deal with software development. This, however, is not the standard under 35 U.S.C. § 103(a).

It is respectfully submitted that although a reference need not expressly teach that the disclosure contained therein should be combined with another, the showing of combinability, in

11

whatever form, must nevertheless be "clear and particular." *Winner International Royalty Corporation v. Ching-Rong Wang*, 53 USPQ2d 1580, 1586-87 (Fed. Cir. 2000).

The Office Action only provides general conclusory statements and fails to explain how including the object sharing of *Almond* would optimize the system of *Wright*. For example, *Wright* already discloses an integrator that manages complexity and change. The Office Action does not explain why replacing or adding *Almond's* development system for managing different versions of the objects would optimize the integrator of *Wright* and the control of the machines in an object oriented control system of *Wright*.

It is Applicant's position, that there is no motivation to add a system for different versions of the object as disclosed by *Almond* to the system of *Wright* that discloses inheritance of objects. Specifically, in *Wright*, there is no disclosure or suggestion of modifying the existing objects for other development tools. Accordingly, there is no motivation to include a system that would allow development tools to access object on the server and modify it for other development tools.

Moreover, in *Wright*, objects are inherited for higher level objects, thereby simplifying programming for the user (col. 6, lines 20 to 39). Accordingly, adding *Almond's* disclosure of version control would change the principle operation of the system in *Wright* and would unjustifiably complicate the design of *Wright's* CNC machine tool control system. For at least these exemplary reasons, it is respectfully submitted that the references are not combinable and there is no motivation to combine the references in the manner suggested by the Examiner.

The Examiner further alleges that one of ordinary skill in the art would have been motivated to combine *Almond* and the *APA* to provide progressive object versions in a multi-tool environment (*see* page 8 of the Office Action).

The Office Action, however, fails to account for the different nature of the *APA* and *Almond*. *Almond* relates to various programming languages such as Powersoft, PowerBuilder and Visual C++ that can access the same objects (software modules) by having different local references to the same object or modify the object on the server via checkout procedure. The *APA*, on the other hand, is related to creating a control system for external machines (*i.e.*, to numeric controllers and the like). This type of control system is a different creature from the traditional object oriented programming such as the one disclosed in *Almond*. For example, this is evidenced by the disclosure of *Wright*, which designs a separate complex system to obtain just one aspect of the object oriented programming *i.e.*, inheritance. To adapt the system of *Almond* to the disclosure of the *APA* would require a substantial reconstruction and redesign of elements in the *APA* as well as changes in the principle operation of the *APA*.

Furthermore, it appears that in the rejection of record, the primary reference is *Almond*. Accordingly, there is no motivation to add various program generating tools of the *APA* to the system of *Almond* that deals with version control. Since *Almond* is unrelated to controlling external machines, there is no suggestion or motivation to add the program generating tools for designing a control system for the external machines. Furthermore, if the system of *Almond* was to include the program generating tools of the *APA*, *Almond's* version control system would need to be redesigned to provide different, complex interfaces for these program generating tools.

13

In short, it is respectfully submitted that one of ordinary skill in the art would not have been motivated and could not have combined these very different references (*Almond*, *Wright*, and the *APA*), which address different problems and are very different creatures of computer science.

In addition, the combined teachings of the references fail to discloses at least the following features set forth in the independent claim 1: "wherein the objects are shared by said program generation tools for generating the programs by transferring the variable name and the attribute data definitions corresponding to the object into each respective program generation tool that shares the object."

That is, the present invention relates to a control system that may include a programmable controller, a display device, a supervisory computer, and so on. These devices control external machines such as equipment in a manufacturing facility. Programs for these devices are drafted separately due to significant differences in the design and generation of these programs but the objects (various data and variables that relate to an external machine) used in these programs are the same.

For example, a valve of an external machine needs to be controlled (by a programmable controller), monitored (on a display device), and shut down when a serious fault is detected (by the supervisory computer when the valve does not open, for example). Accordingly, the same objects are used in the programs of these various devices. In the conventional techniques, the objects have to be drafted separately for each different type of program.

In the present invention, on the other hand, the data is shared by the tools that generate

these various programs. That is, the objects that include a name and attributes such as

input/output variables and so on are shared between these different generation tools. In

particular, the content of the shared object is transferred or uploaded into each generation tool

that uses the object and a table is maintained at the server to notify the server that stores the

definitions when the object is modified so that other tools can update the contents of the object.

The Examiner alleges that when clients connect to the server to manipulate an object,

they share the object version (alleged object name). The Examiner further alleges that the client

share an object name between the server and the client (*see* page 4 of the Office Action).

To begin, it is respectfully noted that version number cannot be equated to an object

name at least because names are unique *i.e.*, two objects will not share the same name. On the

other hand, version numbers are commonly shared amongst objects *i.e.*, each object may have

versions 1, 2, and 3. Accordingly, it is respectfully noted that a version number cannot anticipate

an object name. Moreover, it is noted that claim 1 requires that the objects are shared <u>by the</u>

<u>program generation tools</u> and <u>not between the development tool</u> (alleged program generation

tool) <u>and the server</u> (alleged the data sharing unit).

Furthermore, in *Almond*, there are two methods of accessing an object. One method is to

have multiple names pointing to a single storage location of the object to provide multiple uses

of the object for read access for example (col. 3, lines 11 to 17 and col. 6, lines 52 to 61).

Another method is to check out the object. In order to check out the object, a lock is placed on

the object *i.e.*, other tools can no longer share the object (col. 28, line 41 to col. 30, line 37).

Then, the object data is transferred to <u>local variables</u> for modification, as is common with object-oriented programming (col. 29, lines 30 to 31 and col. 30, lines 48 to 55).

Accordingly, it is respectfully submitted that in *Almond*, <u>an object name is not shared</u> <u>between various development tools</u>. That is, during checkout, local variable are created and during the possible read access multiple names are provided to point to the storage location of the object.

Moreover, it is respectfully submitted that in *Almond*, <u>the object name and its attributes</u> <u>are not transferred to each development tool that share the object</u>. On the contrary, in *Almond*, the object remains in the server and is only transferred to the client during the checkout procedure. However, during this checkout procedure, a lock is placed on the object to prevent the object from being shared or accessed by other clients *i.e.*, development tools.

Accordingly, *Almond* fails to disclose or suggest "wherein the objects are shared by said program generation tools for generating the programs by transferring the variable name and the attribute data definitions corresponding to the object into each respective program generation tool that shares the object," which lacks having an object name shared and which lacks sharing the objects by transferring the object name and its attributes into each respective development tool. *Wright* and the *APA* do not cure the deficient teachings of *Almond*.

Therefore, for at least these exemplary reasons, claim 1 is patentable over the combined teachings of *Almond, Wright*, and the *APA*. It is appropriate and necessary for the Examiner to withdraw this rejection of claim 1.

Claims 3, 4, 11, and 14 are patentable at least by virtue of their dependency on claim 1.

Independent claim 5 recites features analogous to the ones argued above with respect to claim 1. Accordingly, analogous arguments are applicable to claim 5. For at least similar exemplary reasons, therefore, claim 5 is patentable over the combined teachings of *Almond*, *Wright*, and the *APA*.

In addition, claim 5 recites: "wherein one of the program generation tools performs data definition of a variable name and attribute data corresponding to an object in each of said devices forming part of the control system for controlling the group of machines." The Examiner alleges that *Almond* discloses a development tool creating an object and storing it on the server for sharing (*see* page 9 of the Office Action) and that *Wright* discloses "in each of the devices forming part of the control system for controlling the group of machines" (*see* page 7 of the Office Action). These grounds of rejection are not understood.

Almond only discloses developing and storing an object into the proper position on the meta-model (col. 2, line 58 to col. 3, line 2) and fails to disclose or suggest any development tool that would generate an object in each device that forms the control system. *Wright* fails to cure the deficient teachings of *Almond*. That is, *Wright* only discloses that an object may be inherited by other tool objects (col. 6, lines 29 to 40). *Wright*, however, fails to disclose or suggest a tool that would define an object in each device forming the control system. The *APA* fails to cure the deficient teachings of *Almond* and *Wright*. For at least these addition exemplary reasons, claim 5 is patentable over the combined teachings of *Almond*, *Wright*, and the *APA*.

Almond, Wright, and the APA in view of Chou

Claims 2 and 10 are rejected under 35 U.S.C. § 103(a) as being unpatentable over

Almond, Wright, and the *APA* in view of "Versions and change notification in an object-oriented

database system" by Chou et al. (hereinafter "*Chou*"). Claim 2 depends on claim 1 and claim 10

recites features analogous to the ones argued above with respect to claim 1. Accordingly, claim

10 is patentable over the combined teachings of *Almond, Wright,* and the *APA. Chou* is only

cited for its teaching of change notifications (*see* pages 10 and 11 of the Office Action) and as

such fails to cure the deficiencies in the teachings of *Almond, Wright,* and the *APA.*

Accordingly, claims 1 and 10 are patentable over the combined teachings of *Almond, Wright,*

APA, and *Chou.* Claim 2 is patentable at least by virtue of its dependency on claim 1.

Almond, Wright, and the APA in view of Carter

Claims 6, 8, and 9 are rejected under 35 U.S.C. § 103(a) as being unpatentable over

Almond, Wright, and the *APA* in view of U.S. Patent 5,907,705 to Carter (hereinafter "*Carter*").

Claims 6, 8, and 9 depend on claim 5. As noted above, the combined teachings of

Almond, Wright, and the *APA* do not suggest the unique features of claim 5. *Carter* is only cited

for its teachings of notifications and storing data (*see* pages 12 and 13 of the Office Action) and

as such fails to cure the deficiencies in the teachings of *Almond, Wright,* and the *APA.*

Accordingly, claim 5 is patentable over *Almond, Wright,* the *APA,* and *Carter,* taken alone or in

any conceivable combinations. Claims 6, 8, and 9 are patentable at least by virtue of their

dependency on claim 5.

Almond, Wright, and the APA in view of Carter and Levine

Claim 7 is rejected under 35 U.S.C. 103(a) as being unpatentable over *Almond, Wright,*

the *APA*, and *Carter* in view of "Linkers & Loaders" by Levine (hereinafter "*Levine*").

Claim 7 depends on claim 5. As noted above, the combined teachings of *Almond,*

Wright, the *APA*, and *Carter* do not suggest the unique features of claim 5. *Levine* is only cited

for the detection of overlap (*see* page 13 of the Office Action) and as such fails to cure the

deficiencies in the teachings of *Almond, Wright,* the *APA*, and *Carter*. Accordingly, claim 5 is

patentable over *Almond, Wright,* the *APA, Carter*, and *Levine*, taken alone or in any conceivable

combinations. Claim 7 is patentable at least by virtue of its dependency on claim 5.

Almond, Wright, and the APA in view of Atkins

Claims 15-17 are rejected under 35 U.S.C. § 103(a) as being unpatentable over *Almond,*

Wright, and the *APA* in view of a newly found reference, U.S. Patent 5,734,902 to Atkins et al.

(hereinafter "*Atkins*").

Claims 15-17 depend on claim 1. As noted above, the combined teachings of *Almond,*

Wright, and the *APA* do not suggest the unique features of claim 1. *Atkins* is being cited only for

its disclosure of a table storing nodes that use a particular object (*see* pages 14 and 15 of the

Office Action) and as such fails to cure the deficiencies in the teachings of *Almond, Wright,* and

the *APA*. Accordingly, claim 1 is patentable over *Almond, Wright,* the *APA*, and *Atkins*, taken

alone or in any conceivable combinations. Claims 15-17 are patentable at least by virtue of their

dependency on claim 1.

In addition, the Examiner alleges that one of ordinary skill in the art would have been

motivated to incorporate the used-by table of *Atkins* into the system of *Almond* to "maintain

19

concurrency of the shared objects" (*see* page 14 of the Office Action). This reasoning, however, disregards *Almond's* disclosure of version control, where each version of the object is different. That is, the purpose of *Almond's* invention is to integrate various versions of the object into a single schema *i.e.*, main project (col. 2, line 38 to col. 3, line 53) and not to share the various versions of the object by downloading objects into the client *i.e.*, creating multiple copies of various versions of the object. In *Almond*, the user may access the object via a pointer or the object can be checked out for modification.

Specifically, in *Almond*, the object is stored in a central location and is accessed by multiple clients via a pointer. When the object is checked out, a lock is placed on the object preventing other clients from accessing the object. In *Almond*, however, the objects are not downloaded into various clients so as to create multiple copies. Accordingly, *Atkin's* used-by table is of no use to the system of *Almond*. Since in *Almond*, there are no multiple copies of various versions of the object, there is no problem of concurrency. Accordingly, one of ordinary skill in the art would not have combined *Atkins* with the system of *Almond*. For at least this additional exemplary reason, claims 15-17 are patentable over the combined teachings of *Almond*, *Wright*, *APA*, and *Atkins*.

IV.   Conclusion

In view of the above, reconsideration and allowance of this application are now believed to be in order, and such actions are hereby solicited. If any points remain in issue which the Examiner feels may be best resolved through a personal or telephone interview, the Examiner is kindly invited to contact the undersigned attorney at the telephone number listed below.

AMENDMENT UNDER 37 C.F.R. § 1.116
U.S. Appln. No. 09/936,160
Attorney Docket No.: Q65858

The USPTO is directed and authorized to charge all required fees, except for the Issue

Fee and the Publication Fee, to Deposit Account No. 19-4880. Please also credit any

overpayments to said Deposit Account.

Respectfully submitted,

SUGHRUE MION, PLLC
Telephone: (202) 293-7060
Facsimile: (202) 293-7860

WASHINGTON OFFICE
23373
CUSTOMER NUMBER

Nataliya Dvorson
Registration No. 56,616

Date: June 28, 2006

Attorney Docket No.: Q65858

21